NWG/RFC# 581 DHC JBP 1-NOV-73 18:25 19860
Corrections to RFC 560: Remote Controlled Transmission & Echoing TELNET Option

Postel 19860

Network Working Group
Request for Comments: # 581

D. Crocker (UCLA-NMC)
J. Postel (MITRE-TIP)
2 NOV 73

NIC # 19860

References: RFC 560, RFC 563
Categories: Protocols, TELNET, RCTE

RECEIVED
NOV 9 1973

1

Corrections to RFC 560
Remote Controlled Transmission & Echoing TELNET Option

1a

[This RFC contains corrections to RFC 560 (NIC -- 18492,) which described the Remote Controlled Transmission and Echoing TELNET Option. A completely updated version of 18492 has been journalized and will be included in the Protocols Notebook. These new specifications for RCTE are in NIC document (19859,).] 2

Section 1 of the RCTE Option specification (18492,2a:gy) was supposed to include the name and code for the option. The code was accidentally left out. That statement should read: 3

RCTE 7 3a

Section 2 should include the End of Subnegotiation Parameter, at the end of the subnegotiation parameter specification (18492,2b5:gy). All examples in the option specifications, showing RCTE SB commands, should also show the IAC SE parameter. (The revised RCTE specifications have been so changed.) Section 2 should be changed so that it reads: 4

IAC SB RCTE <cmd> [BC1 BC2] [TC1 TC2] IAC SE 4a

The sample scenario, in Section 5.D (18492,2e4:gy), should be modified to reflect the kind of assynchrony of events that can occur with the RCTE protocol. The updated RCTE specifications (in -- 19859,1e4:gy) now reflect this. 5

In RFC 563 (18755,) John Davidson criticizes RCTE's apparent failure to allow Net I/O and server computation to overlap. 6

I agree with John's criticisms and feel that the following should fix the problem: 7

NWG/RFC# 581 DHC JBP 1-NOV-73 16:25 19860
Corrections to RFC 560: Remote Controlled Transmission & Echoing TELNET Option

1. Change 5.A (18492,2e1) 7a

from: 7a1

Overview of Interaction 7a1a

to: 7a2

Overview of User Terminal Printing Action & Control 7a2a

2. Change 5.B.5.a (18492,2e2e1) 7b

from: 7b1

A Transmission character is one which REQUIRES the User Host to transmit all text accumulated up to and including its occurrence. (For Net efficiency, User hosts are DISCOURAGED from sending before the occurrence of a Transmission character). 7b1a

to: 7b2

A Transmission character is one which RECOMMENDS that the Using Host transmit all text accumulated up to and including its occurrence. (For Net efficiency, Using hosts are DISCOURAGED from sending before the occurrence of a Transmission character, as defined at the moment the character is typed). 7b2a

3. Change 5.B.5.b (18492,2e2e2) 7c

from: 7c1

A Break character has the effect of a Transmission character, but also causes the Using host to stop its print/discard action upon the User's input text, until directed to do otherwise by another IAC SB NOTE <cmd> IAC SE command from the Serving host. Break characters therefore define printing units. "Break character" as used in this document does NOT mean Telnet Break character. 7c1a

to: 7c2

A Break character REQUIRES that the Using host transmit all text accumulated up to and including its occurrence and also causes the Using host to stop its print/discard action upon the User's input text, until directed to do otherwise by another IAC SB RCTE <cmd> IAC SE command from the Serving host. Break characters therefore define printing units. "Break character" as used in this document does NOT mean Telnet Break character. 7c2a

4. Change 5.B.6 (18492,2e2f) 7d

from: 7d1

Input from the terminal is (hopefully) buffered up to the occurrence of a Transmission or Break character; and the input text is echoed or not echoed, up to the occurrence of a Break Character. The most recent RCTE command determines the echo, Transmission and Break actions. 7d1a

to: 7d2

Input from the terminal is (hopefully) buffered into units ending with a Transmission or Break character; and echoing of input text is suspended after the occurrence of a Break Character and until receipt of a Break Reset command from the Serving host. The most recent RCTE Break reset command determines the Break actions. 7d2a

5. Change 5.C.4 (18492,2e3d) 7e

FROM; 7e1

A severe (User) site-dependent problem will be buffering type-ahead input from the terminal. It is possible, especially in the case of TIPS, that the input buffer will overflow often. If the receiving (serving) host will permit, the accumulated text should be transmited at this point. If the text cannot be transmited and further typing by the user will result in lost text, the user should be notified. 7e1a

to: 7e2

Buffering Problems and Transmission vs. Printing Constraints: 7e2a

There are NO mandatory transmission constraints. The Using host is allowed to send a character a time, though this would be a waste of RCTE. The Transmission Classes commands are GUIDELINES, so deviating from them, as when the User's buffer gets full, is allowed. 7e2a1

Additionally, the Using host may send a Break Class character, without knowing that it is one (as with type-ahead). 7e2a2

The problem with buffering occurs when printing on the user's terminal must be suspended, after the user has typed a currently valid Break Character and until a Break Reset command is received from the serving host. During this time, the user may be typing merrily along. The text being typed may be SENT, but may not yet be PRINTED. 7e2a3

The more standard problem of filling the transmission buffer, while awaiting an ALLOC from the Serving host, may also occur, but this problem is well known to implementors and in no way special to RCTE. 7e2a4

In any case, when the buffer does fill and further text typed by the user will be lost, the user should be notified. 7e2a5

6. And add 5.C.5, 5.C.6, 5.C.7, 5.C.8, and 5.C.9 as follows: 7f

(5) The Serving and Using hosts must carefully synchronize Break Class Reset commands with the transmission of Break characters. Except at the beginning of an interaction, the Serving host MAY ONLY send a Break Reset command in response to the User host's having sent a Break character as defined at that time. This should establish a one-to-one correspondence between them. (A <cmd> value of zero, in this context, is interprested as a Break Classes reset to the same class(es) as before.) The Reset command may be preceded by terminal output. 7f1

(6) Text should be buffered by the User host until the user types a character which belongs to the transmission class in force at THE MOMENT THE CHARACTER IS TYPED. 7f2

(7) Transmission Class Reset commands may be sent by the Serving host at ANY TIME. If they are frequently sent separate from Break Class Reset commands, it will probably be better to exit from RCTE and enter regular character at a time transmission. 7f3

(8) It is not immediately clear what the Using host should do with currently buffered text, when a Transmission Classes Reset command is received. The buffering is according to the previous Transmission Classes scheme. 7f4

The Using host clearly should NOT simply wait until a Transmission character (according to the new scheme) is typed. 7f4a

Either the buffered text should be rescanned, under the new scheme; 7f4b

Or the buffered text should simply be sent as a group. This is the simpler approach, and probably quite adequate. 7f4c

(9) It is possible to define NO BREAK CHARACTERS except TELNET commands (IAC ...). This might actually be useful, as in the case of tranmsitting on carriage-return, with the Using host echoing (a controlled half-duplex). 7f5

Having the using host send a Telnet Command will allow the serving host to know when he may reset the Break classes, but the mechanism is awkward and probably should be avoided. 7f5a